HUMAN POSE RECOGNITION USING NEURAL NETWORKS, SYNTHETIC MODELS, AND MODERN FEATURES

By

MICHAEL KRUIS

Bachelor of Science in Electrical Engineering

Oklahoma State University

Stillwater, OK

1994

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
July, 2010

HUMAN POSE RECOGNITION USING

NEURAL NETWORKS, SYNTHETIC MODELS,

AND MODERN FEATURES

Thesis Approved:

Dr. Guoliang Fan

Thesis Adviser

Dr. Damon Chandler

Dr Martin Hagan

Dr. Mark E. Payton

Dean of the Graduate College

ACKNOWLEDGMENTS

I would like to thank my Advisor, Dr. Gouliang Fan for his help and guidance in the work presented in this thesis. Without Dr. Fan's assistance, I would never have made it this far in my degree.

I would also like to thank all the members of Visual Computing and Image Processing Lab (VCIPL), especially Xin Zhang for Human Eva ground truth data and additional assistance.

I would like to thank the committee members Dr. Damon Chandler, and Dr. Martin Hagan for taking the time and effort to review my thesis. Dr Hagan was particularly helpful in my implementation of neural networks.

Last but certainly not least, I would like to thank all the friends and family, who made my time in graduate school not only possible, but also enjoyable.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

CHAPTER 1

INTRODUCTION

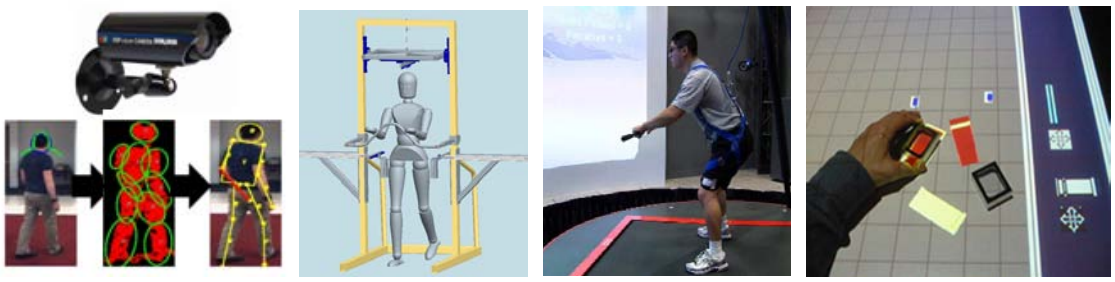
1.1 Motivation

Computer vision has made great strides in the last few decades [18]. In general the ultimate goal of computer vision systems is to design and implement artificial vision systems that can process images like the human visual system. Humans can perceive the information contained in the 3D world and 2D images without effort, computers require considerable effort to achieve even limited approximation of this simple visual task. In most cases computer vision still remains a collection of diverse studies motivated by specific applications. Researchers have presented a wide range of specific problems in computer vision. Most of their solutions experience difficulties when used in the real world limiting their usefulness. The motivation for this research is to more closely mimic the human visual system and utilize the latest advances in object recognition and machine learning.

One of the most important tasks that the human visual system performs is human pose recognition or estimation [10],[11],[12]. Human pose recognition involves identifying the 3D pose of a human body from a 2D image. 3D pose data is obtained by the process of motion capture. Traditionally motion capture requires markers attached to the body joints. These systems have some major flaws as they are obtrusive,

expensive, and impractical in applications in which the observed humans are not cooperative. The majority of images do not contain convenient markers of body joints. As such many applications especially in surveillance and human computer interaction would benefit from a markerless solution. Markerless human pose recognition is an important, yet challenging computer vision task. Pose recognition is intertwined in a series of computer vision tasks that involve detecting, segmenting and tracking humans in images and video. This thesis does not deal with human detection, segmenting or tracking. We limit ourselves to estimating human body pose from markerless images. An additional task that follows pose recognition is action recognition. Action recognition, interpreting movement over time, is not discussed in this thesis.

Human pose recognition has many important applications. It is a critical part in human-computer interaction, surveillance, safety control, sports medicine, sports rehabilitation, animation, markerless motion capture, indexing video libraries, and many other applications. Figure 1 shows a few of the applications of human pose recognition.



(a) (b) (c) (d)

Figure 1: Applications for human pose recognition (a) Surveillance (b) Medical (c) Sports training (d) Human computer interaction (Images from (a) http://homesecuritycameratips.com/wp-content/uploads/2010/03/outdoor-camera.jpg http://staffnet.kingston.ac.uk/~ku33185 /MEDUSA/pipeline.jpg (b) http://www.northeastern.edu/nupr//images/RGR_Trainer.png (c) http://engineeringworks.tamu.edu/episodephotos/12-16-09-high-tech-rehabilitation.jpg (d) http://www.fjeld.ch/hci/navigation_tools.jpg).

With the increase in computer speed and complexity and the internet availability of image and video data the demand for advanced automated human pose recognition algorithms has increased. Most surveillance systems are used to monitor humans. Automated human detection, tracking, segmentation, and human motion analysis are key desired components of the new generation video surveillance systems. These systems will be able to detect and determine the activities, of the individuals, in the videos they process as in Figure 1(a). Human pose recognition is also proving to be invaluable in athletic and medical rehabilitation as in Figure 1(b),(c). An athlete's performance can be greatly improved with motion analysis, by indicating areas of motion inefficiency, which can be improved. Automatic video based human motion analysis algorithms can quickly provide accurate motion and gait information, which will make diagnosis of possible medical problems much simpler. With the increased use of robots, in everyday life, the use of human pose recognition will be of ever increasing use in human computer interaction Fgure1 (d). These systems require accurate understanding of human pose for computer systems and humans to safely and meaningfully interact.

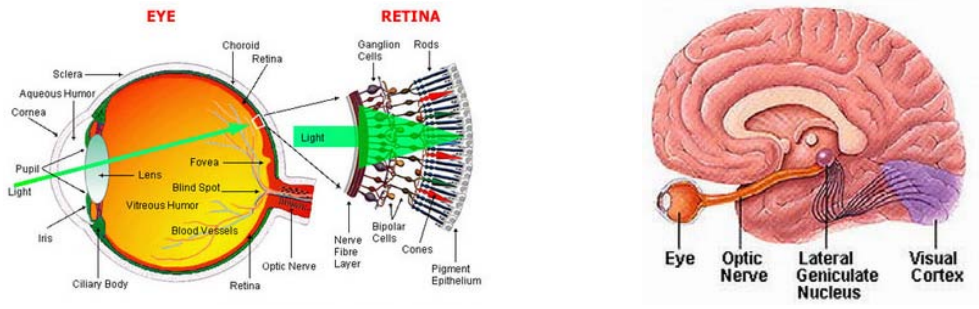
1.2 Research Goals and Challenges

1.2.1 Overview

The goal of this research is to utilize computers to mimic the human visual system in the task of pose recognition. The human visual system, Figure 2, has two main components the eyes and the brain, connected by the optical nerve. Nerve cells in the retina of the eye convert the light signals from the outside world into electrical impulses.

These electrical impulses travel down the optic nerve, and are processed in the lateral geniculate nucleus and the visual cortex of the brain.

The input into our artificial visual system is not light, from the outside world, but instead 2D digital images. The function of the eyes retina is to filter the important information of the real world, and send that information to the brain. Image features filter the important information from digital images. For our artificial system we experiment with a wide variety of features to extract pose information from images. Many different features have been developed in the computer vision field, to solve a wide variety of problems. We are interested in features that aid in human pose recognition. To compare a diverse set of features, traditional and newer features are utilized in our artificial vision system. The features can be looked as simulating the function of the eye, but instead of processing light information they process digital images. For our artificial brain we have chosen artificial neural networks. Of machine learning techniques, artificial neural networks most closely resemble real biological brains.



(a) (b)

Figure 2: Human visual system (a) Eye (b) Brain (Images from (a) http://ghazwaaldoori.com/images/structure%20of%20rhodopsin%20picture.jpg (b) http://mthago.files.wordpress.com/2010/02/eye-brain1.jpg).

1.2.2 Human Visual System

The human visual system has had millions of years to evolve. Our visual system is the primary tool we use to perceive the world around us. As one of the most complex and efficient visual system we know of, there is much to be learned from studying it [19]. One way, of interest to this research, that the human eye efficiently process information is by highlighting certain incoming information. A region of the eye called the fovea contains the majority of the retinas nerve cells, and is responsible for sharp central vision which is essential for any activity where visual detail is of primary importance. The eye is also designed to rapidly alter its direction, to focus on interesting points in its visual field. Another feature of the eye that is of interest, for this research, is its sensitivity to contrast as oppose to absolute luminance. The human eye is sensitive to edges and sharp changes in contrast. We perceive the world similarly regardless of huge changes in illumination over the day or from place to place. Our artificial visual system should incorporate these features, to more closely match the human visual system.

Even though computer vision has only been around for a few decades both of the above attributes have been incorporated into modern feature algorithms. The task of object recognition in digital images has generated a wide variety of modern features. Features have been developed to systematically deconstruct image gradients and their orientation. These features have been used to successfully detect humans in images [3]. Another class of features, interest point features [8],[9], has been designed to find and describe in detail interesting points in digital images. These interest point features mimic the eye by focusing the visual system. First these features detect interesting points in an image that are consistent over a wide range of transformations. Second these features

describe only a region around the interest point. Comparing the interest point feature descriptors allows for accurate object recognition in a wide range of images. With these two recent advanced features we are seeing an increase in sophistication of our ability to characterize digital images. In this thesis examples of both of these advanced features are utilized. These and existing features are compared in the task of human pose recognition, and their usefulness is discussed.

1.2.3 Challenges in Human Pose Recognition

Human pose recognition creates a great many problems due to the nature of the human body and its representation in digital images. The following are a list of the major problems with humans as represented in digital images

1. The human body is a highly articulated 3D object with a wide range of possible poses.
2. The human body also shows a wide range of shape differences.
3. Humans in digital images wear a wide variety of clothing and accessories with differences in shape, color, and texture.
4. Digital images show a wide range of backgrounds, lighting conditions, partial views, or multiple occluded views.
5. The relationship between pose and observation is not direct or single valued, due to variation between people in shape and appearance and different camera viewpoint and environment the same pose can have many different observations.
6. Also different poses can result in the same observation, since observation is a 2D projection, information is lost.

All these factors, and more, make human pose recognition from digital images arguably one of the most challenging areas of study in computer vision.

1.3 Research Objectives and Methodology

1.3.1 Problem Formulation

Our goal is human pose recognition, with digital images and artificial methods that simulate the human visual system. To create a system that can be effectively utilized to compare a wide range of features a series of simplifications to the problem are implemented. These simplifications will then be removed in future work to generalize the solution.

We begin by utilizing five commercially available synthetic models for training and testing illustrated in Figure 3. With synthetic models we can control the view and the pose. Commercial motion builder software allows us to generate video of each model in a wide range of activities and viewed from any location. These models allow for a high degree of accuracy in the 3D location of body parts as the model performs an activity. This degree of freedom in choosing pose and camera location greatly increases the consistency and usefulness of the input data. From the video of these models we can obtain digital images associated with specific poses.

Figure 3: Five 3D synthetic models first from MotionBuilder Clip of Art and the others are from www.axyzdesign.com

Instead of utilizing the original image we describe the image utilizing only the silhouette of the human. Since we focus on recovery of human pose of a person we would like to generalize over image variations as mentioned earlier. Part of this generalization can be handled in the image domain by extraction image descriptors like silhouettes rather than using the original image. In other words, we do not need complete knowledge about how a model appears in the image domain. The silhouette contains the 2D shape information and has been used extensively in pose recognition. Our synthetic model video images can easily be converted into silhouette images. This simplification greatly reduces the vector size and complexity of our image representation. The disadvantage of utilizing silhouettes is that all depth information in the image is lost.

The models in Figure 3 are all facing to the front. From this view, the pose is a lot harder to determine than say the side view. This view is also very similar to the view of the model from the back. For this reason the view angle is made discrete around the model. Figure 4 indicates the 12 view directions as viewed from above the model looking down. Each view direction is considered separately and pose recognition is done separately for each of the 12 views around the model. The models in Figure 3 would be in view #1 the left side view is view #4 and so on. This solves a great deal of ambiguity generated between image and pose.

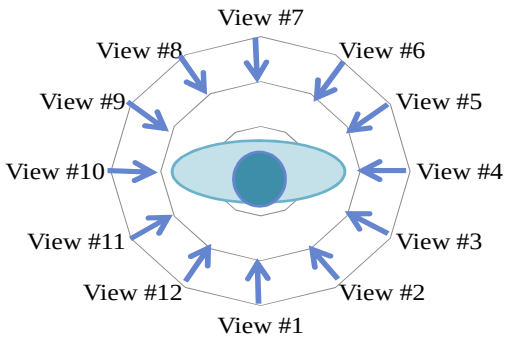


Figure 4: View positions around human model as viewed from above.

We further limit the pose by only looking at the walking pose. Figure 5 shows the mean silhouette of the ten poses in the full walking cycle, for one of the synthetic models at view #4. The full walking cycle can be further simplified by only using the half walking cycle. Note that the silhouettes of the first five poses are similar to the last five poses. In reality these poses are quite different with the former representing the first half of the walking cycle, with left leg in front, and the latter representing the second half of the walking cycle, with right leg in front. While silhouettes are a powerful tool for analyzing pose they do introduce ambiguities. With respect to silhouettes the first half of the full walking cycle is similar to the second half; both the half cycle and the full cycle are utilized in this research. The half cycle contains the first five poses of the full walking cycle.

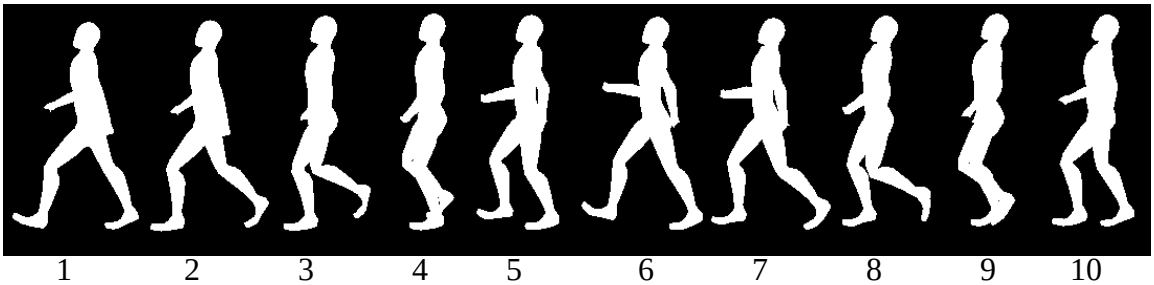


Figure 5: 10 mean poses of the full walking cycle as viewed from view #4.

These four simplifications greatly simplify the problem and allow for extensive testing of a wide variety of features in an efficient manner. Although these simplifications take us away from a general solution to the problem of human pose recognition they are a practical solution to the problem of how to study the effectiveness of different features. Our focus is more on the image feature algorithm and their relative strengths and weaknesses.

1.3.2 Methodology

The method of evaluation will proceed as indicated in Figure 6. The data will first be separated into training and testing data. Then input features will be extracted from both training and testing data. The neural networks will be trained on the training data. The testing data will then be applied to the trained neural networks. And finally the features will be evaluated for accuracy. As the training and testing continue the advantages and disadvantages of the different features will be discussed. The feature extraction and the neural networks are both implemented in Matlab. Matlab allows for easy visualization and integration of our features and neural networks.

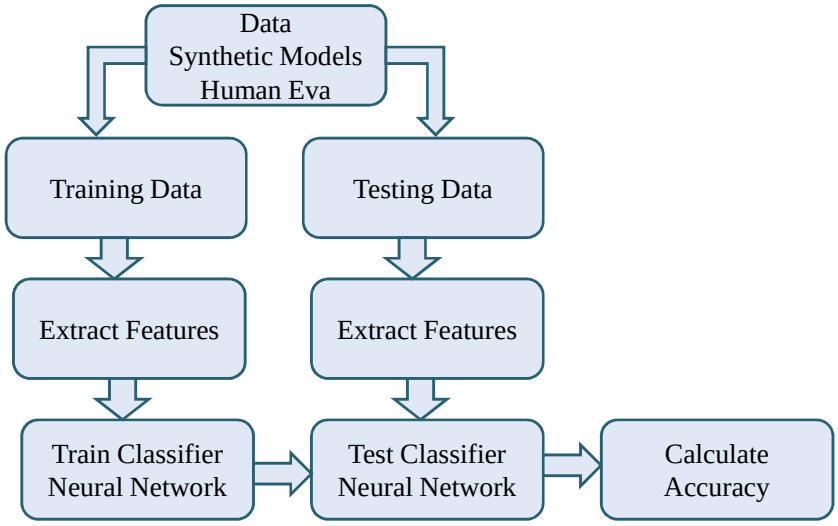


Figure 6: Procedure for the evaluation of features.

1.4 Contribution

The contribution of this research is to investigate the use of artificial neural networks, for human pose recognition, with a variety of modern features. Traditional features from the beginning of computer vision research up to more modern features are

utilized in this research. Recent years have seen advances in features and classification algorithms to simulate the human visual system. These advances often progress on parallel tracks for different applications in computer vision. This thesis attempts to combine the advanced features, from object recognition, and advanced neural networks, from machine learning, and assess their usefulness in the task of human pose recognition.

Also in this thesis new feature vectors algorithms are developed from modern features. A series of feature vectors based on the interest point features are developed and compared with existing features. In this work feature classification accuracy, speed, and memory requirements are explored.

CHAPTER 2

BACKGROUND

2.1 Overview

Currently many new directions of research, in human pose recognition, are being explored such as particle filtering algorithms, top down and bottom up models, and model free approaches. It would be difficult to summarize and discuss all the trends in this area, some comprehensive surveys are discussed for reference. Following the surveys a detailed background of machine learning, features, and neural networks are presented as these are the focus of our approach. In the area of human motion analysis several surveys have been written each with a specific focus. The earlier work of Wang[12] and Gavrila [10] divides 2D pose recognition into approaches with or without the explicit use of shape models. Aggarwal and Chi [11] explores human pose, tracking, and detection. Most approaches divide the human pose into model and model free depending upon whether a-priori information about the object shape is employed. Other more current surveys of human pose recognition, tracking, and human behavior analysis are Forsyth [13] and Poppe [17].

At the same time that human pose estimation has been advancing so has the task of object recognition. A key issue in object recognition is the need for prediction to be invariant to a wide variety of transformations of input images due to scale, translation,

and rotation of the object in 3d space, changes in viewing direction, and distance, and non rigid transformation of the object itself. The selection of and design of features is of great importance in imparting this invariance. Dalal and Trigg in [3] successfully utilized Histograms of Oriented Gradient (HOG's) to detect humans in casual images. And Lowe in [8] and Tuytelaars and Van Gool in [7] developed features, SIFT and SURF, that find points of interest and create local interest points descriptors. These modern features have only recently been utilized in the task of human pose recognition as in Rogez and Ramalingann in [6] which utilizes a sparse set of HOG features with randomized decision trees to successfully detect and estimate the pose of a human in an image.

Pose recognition can be broadly looked at as a multiclass classification problem in machine learning. That involves the assignment of a class label, or pose, to an input object or image. A major focus of machine learning research is to automatically learn to recognize complex patterns and make intelligent decisions based on data; the difficulty lies in the fact that the set of all possible behaviors given all possible inputs is too complex to describe generally. This is certainly the case for the problem of human pose recognition. The following sections first discuss machine learning, features, and then artificial neural networks.

2.2 Generative vs. Discriminative Models

Many machine learning approaches to pose recognition are founded on probability theory and can be broadly characterized as either generative of discriminative. These two approaches can be distinguished according to whether or not the distribution of the image features is modeled, as in generative models, of not modeled, discriminative

models. Generative and discriminative models have very different characteristics as well as complementary strengths and weaknesses but both rely heavily on feature extraction from the input data.

Consider the scenario in which an image described by a vector *Y*, which might be pixel intensities or some set of features extracted from the image, is to be assigned to one pose represented by *X*. From basic probability theory we know that the most complete characterization of the solution is expresses in terms of the set of posterior probabilities. The posterior probability $p_\theta(X|Y)$ can be calculated directly from the vector *Y*. In which the parameters θ are to be learned given a training set of (*Y*,*X*) pairs. Where *Y* is the observation, and *X* is the hidden state. This is the discriminative method. The posterior probabilities can also be calculated using the likelihood and the prior probability. This is the generative model.

$$p_\theta(X/Y) \propto p_\theta(Y|X) \cdot p(Y). \tag{2.1}$$

When we know these probabilities it is straight forward to assign the image *Y* to a particular class to minimize the expected loss. To minimize misclassification we assign *Y* to the class having the largest posterior probability.

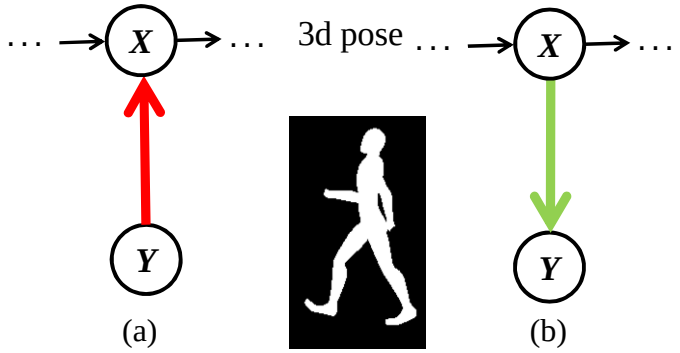


Figure 7: Probabilistic models for human pose recognition (a) Discriminative model (b) Generative models

Generative models, model the observation after all observations and hidden states are learned. Generative models search the pose space for configurations with good image alignment. In this model, Figure 7(b), we go from the 3D pose to the image. The posterior probabilities are calculated utilizing the prior probability and the likelihood.

The advantage of the generative model is that it can handle messy data and partially labeled data. Generative models can generate, from small amounts of labeled data, a large amount of unlabeled data. Generative models can learn a new class independently by learning the conditional density of the new class. Generative models are more stable to noise while discriminative models need to be trained on all combinations of image and pose. Examples of generative models are; Gaussian Mixture Model, Hidden Markov Models, and Markov Random Field.

Discriminative models make no attempt to model the underlying probability distribution, but directly model the posterior probability. Discriminative models focus the computational resources on a given task as indicated in Figure 7(a).

The advantage of discriminative models are that once learned they are faster at making predictions, because generative models need to iteratively search for a solution. All things being equal discriminative models should have better predictive performance because they are trained to predict the class label rather than the joint distribution of input vectors and targets.

The downside of discriminative models, for pose recognition, is that they have to model complex multi-valued 2D to 3D relations. They lack the elegance of the generative models with the calculation of the priors and understanding of the likelihood. In many cases discriminative models feel like ‘black boxes’ that just spit out the answer.

Examples of discriminative models are; support vector machines, conditional random fields, logistical regression, and neural networks.

Although this work makes no contribution to the debate as to the type of model that should be utilized it is important to know the context of this work. The human brain appears to work more as a ‘black box’ or as a discriminative model. In both models the observation, or features, are of importance. In the discriminative model it is what drives the classification. In the generative model it is what is used to build the model of the prior. Due to its simplicity, power, and similarity to biological systems this work utilizes artificial neural networks in a discriminative model.

Separate from the research for pose recognition many advances have been made in the computer vision areas of object recognition and tracking. This work tests the effectiveness of a set of traditional and modern features in a discriminative model utilizing neural networks.

2.3 Features

2.3.1 Overview

In computer vision and image processing the features are used to denote a piece of information which is relevant for solving the computational task related to an application. The concept, of features, is very general and the choice of features in a particular computer vision system may be highly dependent on the specific problem at hand. For this reason a wide range of features should be experimented with for any particular task. The features in our problem are related to human motion in image sequences. The images are first segmented into foreground and background. The features are defined in terms of

boundaries between different image regions, or to properties of such a region. These features are extracted and paired with their 3D pose and used to train our artificial neural networks.

Features can be categorized into two groups based on the representation as shown in Figure 8. The first group of features is the dense, or holistic, representation. The second representation is the sparse, or parts, representation. Ideally a feature vector would contain all the image information but size and computational complexity prohibit effective use of such a vector. In general feature vectors are designed to be easy to extract, small, and have ability to discriminate in the specific application used. Features that are invariant to a wide range of transformations are more discriminative. As computers become more advanced the size and complexity of features has advanced. A large part of feature design, for image processing, is reducing the feature size, fixing the feature size, and increasing the speed of calculation.

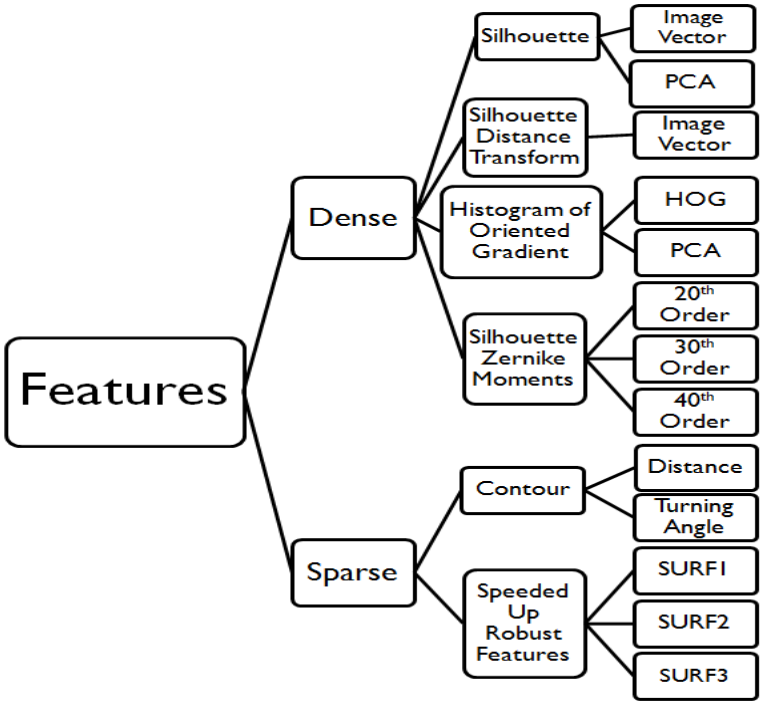


Figure 8: Feature Taxonomy.

2.3.2 Dense Features

Dense features utilize complete image information at the cost of memory or speed of calculation. Dense feature descriptors are obtained over the entire image and each pixel contributes. In general dense features create large feature vectors that have a fixed length. Because every pixel contributes they encode spatial information. These features suffer from increased memory use and added computational time, but have greater discriminative power than sparse features. The dense features utilized are the silhouette image vector, silhouette distance transform image vector, Histograms of Oriented Gradient (HOG's) [3], and Zernike moments [2].

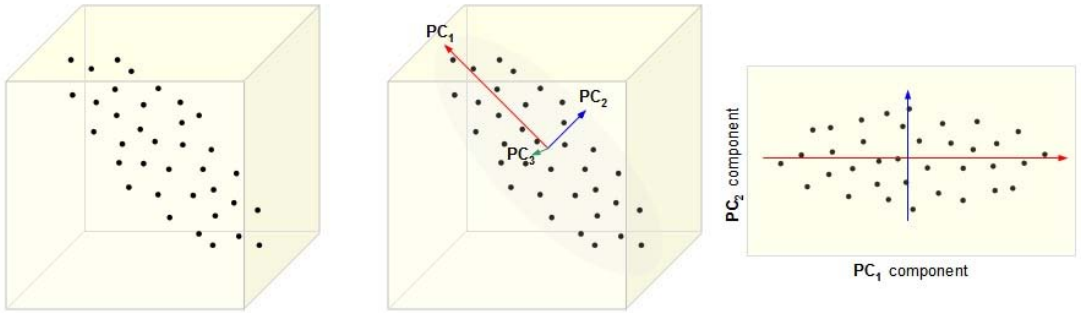
Silhouettes are insensitive to variations in appearance such as color and texture and encode a great deal of information to help recover 3D pose. Silhouettes have been successfully extracted from images when backgrounds are reasonably static. In older studies backgrounds were assumed to be different in appearance from the person. However performance is limited be shadows noisy background segmentation and it is often difficult or impossible to recover pose information due to lack of depth information.

Histograms of Oriented Gradient or HOG's [3] are a dense feature introduced by Dalal and Triggs in 2005 for human detection in casual images. HOG's have shown themselves to be one of the most robust features for the difficult tasks of detecting humans in casual images. Because of the wide variability in appearance due to clothing, articulation, and illumination conditions in casual images of humans, any feature able to accurately detect humans is worthy of consideration in the pose recognition task. This feature has been used in the task of pose recognition as in [6]

Moments are a quantitative measure of the shape of an image. The first order moments are the center of gravity of the intensity image. The second moment, as another example, is widely used and measures the width of an image intensity in one direction. Other moments describe other aspects of a distribution such as how the distribution is skewed from its mean, or peaked. These and higher order moments can be used to accurately describe an image in a very compact form. Hu [1] published the first significant paper on the use of image moments for pattern recognition. Initially using the nineteenth century work on algebraic invariants, Hu derived a set of seven scale, translation, and rotation invariant normalized central moments. Teague [2] observed that the Cartesian moments are in the form of the projection of image intensity onto the non-orthogonal, monomial basis set. Replacing the monomials with an orthogonal basis set (e.g. Zernike polynomials), results in an orthogonal moment set. Zernike moments give full translation, scale, and rotation invariance to any arbitrary order. Image Zernike moments can also be used in image reconstruction. Because of the compact nature of moment features the majority of the research related to moments is concerned with image compression. A review of this literature and additional moments, projections onto additional basis sets, is out of the scope of this thesis.

Reducing the dense features is an intermediary step between dense features and sparse features. The idea is to reduce the size of the dense feature but maintain the discriminative power of the original. In these features all pixels contribute to the calculation of the vector, but dimensional reduction tools are utilized to create a compact representation of the image. Compact representations of images are of great value for computer vision researchers interested in compression. Digital images by their nature

consume a large amount of memory, features that can represent the image in a compact form, are well worth contemplating. The dense features with reduced representations are Principle Component Analysis (PCA) silhouette and PCA HOG features.



(a) (b) (c)

Figure 9: Principle Component Analysis (a) 3D data set (b) Three Principle Components (c) Projection of data onto first two PC's (Image from http://cnx.org/content/m11461/latest /pca.jpg).

Principle Component Analysis PCA is utilized to reduce the size of dense features. PCA is a linear dimensional reduction technique and involves a mathematical procedure that transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible as shown in Figure 9. PCA is applied to both the binary silhouette vector and the HOG vector and treated as separate features. This reduction in feature vector size comes at the price of losing spatial information and the loss of fixed feature vector size. Different principle components are required for each view.

2.3.3 Sparse Features

In the sparse representation a set of local descriptors are obtained as a result of either an interest point detector or some limit on image location. In sparse representation only selected pixels, those near interest points, contribute to the feature. In general sparse features are smaller and easier to calculate, but are not as discriminative as dense features. Sparse features also suffer from not having fixed features size and no spatial information. The sparse features utilized in this thesis work are contour distance and turning angle, and Speeded Up Robust Features [9].

For the contour features the edge of silhouettes are the interest points in our sparse feature. Contours are extracted starting at a fixed point and proceeding around the silhouettes until the original point is reached. As we proceed around a contour two types of basic information can be extracted distance and turning angle. The distance provides a measure of the distance a point is from our fixed starting point and the angle is change in contour angle around the silhouette. The distance and turning angle are sensitive to the length and orientation of extended limbs, and correlates to our notion of shape similarity. These two metrics provide a location and orientation of extremities such as arms and legs.

Extremities are a compact posture representation. Extremities can be utilized for pose recognition by combining the position of the appendages with fixed positions in the silhouette such as the center of mass or top of the head. Many approaches to pose recognition utilize variations of this feature as in [4] [7] [14].

SURF (Speeded Up Robust Features) [9] features are one of a relatively new class of features, interest point features, that incorporate several functions into one feature. In

the first step these features detect interest points in an image. In the second step, local descriptors are created for each of the interest points. In the final step matching is performed with descriptors from other images. Interest point features have greatly advanced the computer vision task of object recognition. Scaled-Invariant Feature Transform or SIFT [8] features are the most popular and one of the first interest point features. Although SURF and SIFT features utilize drastically different interest point detectors, local interest point descriptors and matching algorithms they show similar performance. Through intelligent chose of interest point detector and local descriptors these features are invariant to a wide range of transformations and even partial occlusion. The SURF feature has been optimized to utilize the integral image and box filters which increase the speed of calculation, over SIFT features, by six times. For this reason SURF features are utilized in this thesis.

2.4 Artificial Neural Networks

Neural networks have recently seen resurgence in use and interest both in academics and real world applications [5], in the attempt to achieve autonomous intelligent behavior. Neural networks grew out of a recent shift in the understanding of the function of the brain. It was thought that the human brain functioned as a digital computer with a central processing unit that executed a series of rigid rules. In the neural network understanding, the brain does not function as the central processing unit implementing a set of rules. Neural networks, like the brain, function with many simple elements that act locally to create a global result. Many researchers now believe that the

neural networks paradigm better describes the brains function, and computers are now used to simulate biological neural networks.

Neural networks are artificial simulations of biological brains. The biological brain consists of a large number (approximately 10^{11}) interconnected elements called neurons. The arrangement and strength of neurons, determined by genetics and complex chemical process, establish the function of the neural network. Artificial neural networks seek to emulate this system. With the rise in speed and complexity of modern computers, the ability to solve complex problems with artificial neural networks has become a possibility.

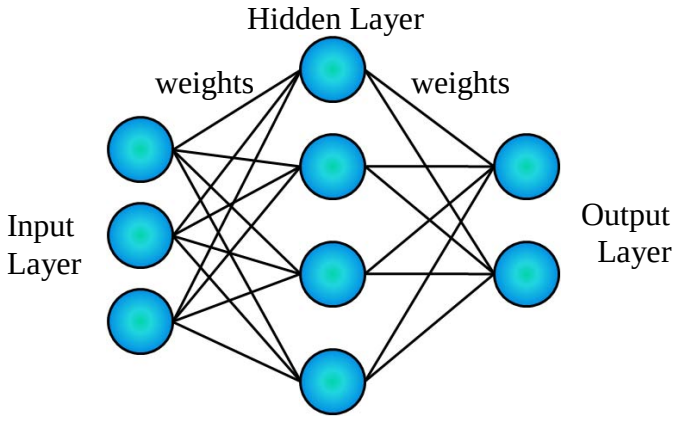


Figure 10: Basic Neural Network.

Artificial neurons are organized in layers as shown in Figure 10. The input layer is the image feature and the output layer is the pose classification. Depending on the strength of connection to each neuron in the hidden layer, the input signal is sent to the next layer. The strength of the connection is called a weight. The value of each neuron in each layer will depend on the weight of the connection and the values of the neurons of the previous layer.

One of the most important features of the brain is its ability to categorize. The human brain is continually presented with a new events and objects. These objects and events are moving, unstable, and unlabeled. Still the brain can distill this chaotic mess of information into categories and representations of objects and events. To construct our artificial visual system we also need the ability to categorize. Our artificial neural network should learn the pose from a set of features.

CHAPTER 3

FEATURES

3.1 Silhouettes and Distance Transform

Silhouettes are one of the oldest and simplest image features utilized for pose recognition. Figure 11(a) shows silhouettes of a synthetic model at four different views and in one pose. Silhouettes have some major advantages and disadvantages. One of the silhouettes advantages is that it contains a great deal of pose information in the form of a

(a)

(b)

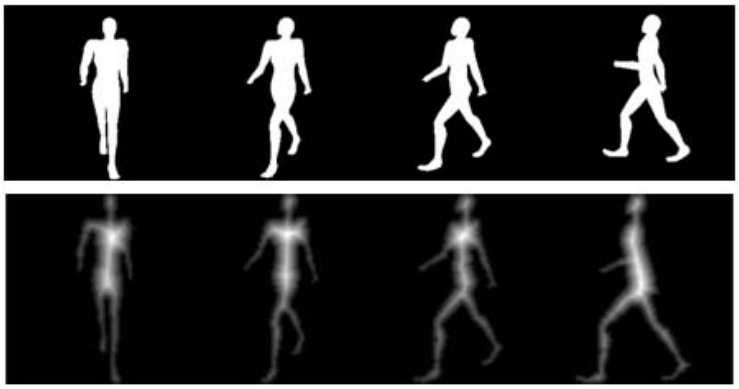


Figure 11: (a) Top row silhouettes (b) Bottom row distance transform of silhouettes

flattened human shape. Another advantage of the silhouette is that there has been much work on background segmentation to extract the human silhouette from digital images and videos. This has lead to the wide spread use of the silhouette for a number of computer vision tasks including pose recognition. Only recently have more modern

features appeared, that incorporates additional image information. One disadvantage, of the silhouette, is that it contains no depth information. This is a problem for human pose recognition in that different poses at different views can produce the same silhouettes.

The distance transform of the silhouette is an attempt to add some depth to the silhouette. Figure 11(b) shows the distance transform of the silhouette in Fig 11(a). Consider a binary image *I*, 0 is black, and 1 is white. The object *K* represents all the white pixels and K' represents all the black pixels. The distance transform is the transformation that generates a map *D*, of binary image *I*, whose value in each pixel *p* is the smallest distance from this pixel to K'.

$$D(p) = \min\{d(p,q) \mid q \in K'\} = \min\{d(p,q) \mid I(q) = 0\}. \tag{3.1}$$

The resulting pixel for the silhouette contains the distance calculated. *D(p)* is called the distance transform of *I*, and *d(p,q)* is generally taken as the Euclidean distance.

$$d(p,q) = \sqrt{(p_x - q_x)^2 + (p_y - q_y)^2}. \tag{3.2}$$

This gives the resulting image a skeletonized look. The regions in the center of the shape retain the highest value much like the bones are at the center of many body regions. This adds additional information to the binary silhouette image with the intent of simulating depth. This depth information from the distance transform is artificial, and is devoid of information about occluded body parts.

The training of the neural networks requires feature vectors of a uniform size for each image pose pair. Constructing a uniform feature vector, with dense features, is relatively simple because their size is fixed. The image pixels of *I* are ordered into one long vector by taking the columns, from left to right, and stacking them on top of each

other. This becomes a very large vector even for a small silhouette image so the original image is reduced to a 110x110 pixel image. The reduced image will lead to a 12,100 number vector for both the silhouette binary image *I* and the distance transform image *D*.

3.2 Histograms of Oriented Gradient (HOG)

The method of HOG's is based on the evaluating of well normalized local histograms of image gradient orientations in a dense overlapping grid. The basic idea is that local object shape and appearance can be characterized well by the distribution of local intensity gradients of edge directions, even without precise knowledge of the corresponding gradient or edge positions. This makes intuitive sense in that the precise location of edge and gradient are not as important as the distribution of the direction.

The first step to implementation of HOG's begins by calculating the gradient of the image Figure 12(a). The most common method is to simply apply the 1D centered, point discrete derivative masks in both the horizontal and vertical directions. Specifically, this method requires filtering the color or intensity data of the image with the following filter kernels:

$$[-1, 0, 1] \text{ and } [-1, 0, 1]^T. \tag{3.3}$$

The second step of HOG calculation involves creating the cell histograms Figure 12(b). Each pixel within the cell casts a weighted vote for an orientation-based histogram channel, or bin, based on the values found in the gradient computation. The cells themselves can either be rectangular or radial in shape, and the histogram channels are evenly spread over 0 to 180 degrees or 0 to 360 degrees, depending on whether the gradient is "unsigned" or "signed". Only rectangular, 8 x 8 pixels, "unsigned" cells, with

gradient magnitude as weight are dealt with in this thesis. These parameters were shown to perform best in human detection experiments [3].

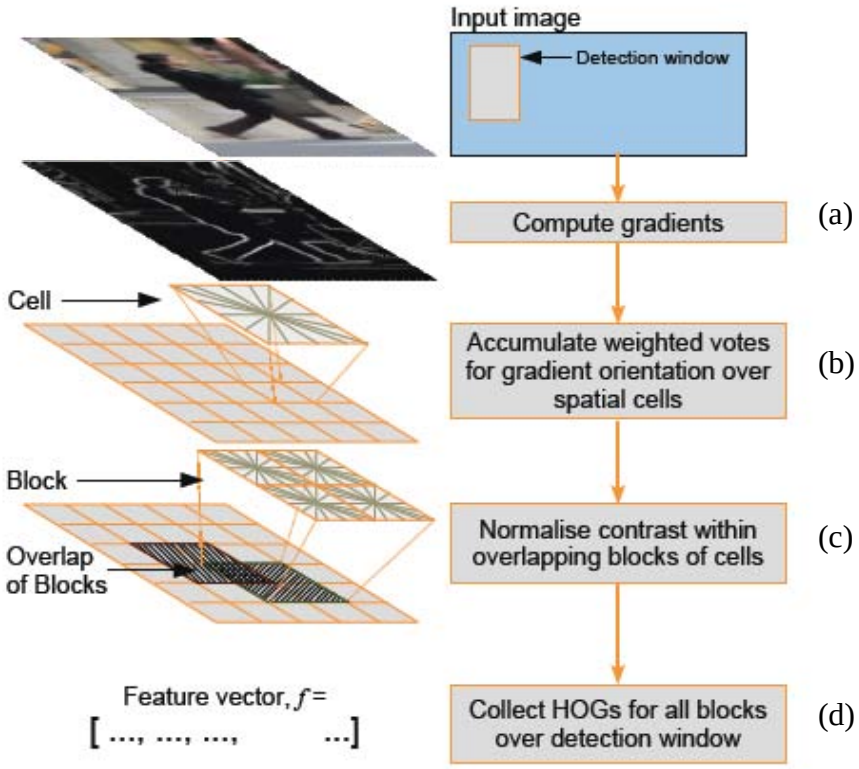


Figure 12: Procedure for extracting HOG feature (Image from www.andrew.cmu.edu/user/ehsiao/old/presentation.ppt*)*

In order to account for changes in illumination and contrast, the gradient strengths must be locally normalized, which requires grouping the cells together into larger, spatially-connected blocks Figure 12(c). L2-norm normalization is utilized for block normalization if v is the feature vector, α is a small constant (.001). The L2-norm v is

$$v = (z_1, z_{2,} \Lambda\ z_n)$$

$$v \rightarrow \frac{v}{\sqrt{\left(\sum_i z_i^2\right) + \alpha^2}} \tag{3.4}$$

The HOG descriptor is then the vector of the components of the normalized cell histograms from all of the block regions Figure 12(d). These blocks typically overlap, meaning that each cell contributes more than once to the final descriptor. The parameters used in this thesis are 8x8 pixel cells with 9 bins, 2x2 blocks, half overlapping. The image used is 112x112 pixels. Each block has 9x4 or 36 bins. With a 112x112 pixel image and half overlapping blocks, 13 blocks can fit along each dimension.

The HOG feature vector is fixed at 6,084 numbers for the 112x112 pixel image and the above parameters. With the HOG feature vector, constructing a uniform feature vector is simple because the size is fixed, for a fixed image size. The vector size can be changed by altering the cell, block, bin, or overlap parameters. In this work the optimal numbers, for human detection, as indicated above were utilized. The Matlab implementation is indicated in [3].

3.3 Silhouette Zernike Moment

Moments are a statistical representation of the intensity image *f(x,y)*. The important properties for moments or any feature are invariance. The basic invariance's, needed for pose recognition, are translation, scale and rotation. Traditionally, moments invariants are computed based on the information provided by the shape. The moments used to construct the moment invariants are defined in the continuous but for practical implementation, digital images, they are computed in the discrete form. Given a image intensity function *f(x,y)*, the regular moments are defined by:

$$M_{pq} = \int\int x^p y^q f(x,y)dxdy \qquad (3.5)$$

Where M_{pq} is the two-dimensional moment of the image intensity function *f(x,y)*. The order of the moment is (*p* + *q*) where *p* and *q* are both natural numbers. For implementation in digital form this becomes:

$$M_{pq} = \sum_{X} \sum_{Y} x^{p} y^{q} f(x, y). \tag{3.6}$$

For translation invariance in the image plane the image center of mass, or centroid, need to be calculated. The co-ordinates of the center of gravity of the image are calculated using equation (3.6) and are given by:

$$\bar{x} = \frac{M_{10}}{M_{00}} \qquad \text{and,} \qquad \bar{y} = \frac{M_{01}}{M_{00}}. \tag{3.7}$$

The central moments can then be defined in their discrete representation as:

$$\mu_{pq} = \sum_{X} \sum_{Y} (x - \bar{x})^{p} (y - \bar{y})^{q} f(x, y). \tag{3.8}$$

The moments are further normalised for the effects of change of scale using the following formula:

$$\eta_{pq} = \mu_{pq} \Big/ \mu_{00}^{\ \gamma}, \tag{3.9}$$

where the normalisation factor: $\gamma = (p + q / 2) + 1$.

Teague observed that the Cartesian moment definition equation (3.8) has the form of the projection of $f(x, y)$ onto the non-orthogonal, monomial basis set. Replacing the monomials with an orthogonal basis set (e.g. Zernike polynomials), results in an orthogonal set of moments this gives full translation, scale, and rotation invariance to the Zernike moment up to any arbitrary order (*p*+*q*). The order of the moment determines the level of detail that is encoded from the image into the moments and also the complexity

of computation as shown in Figure 13. In general there is no easy solution to determining the optimal order for an arbitrary image.

The moments are one of the most compact representations of the image, but the calculation of the moments can be a long tedious process. The long computation time,

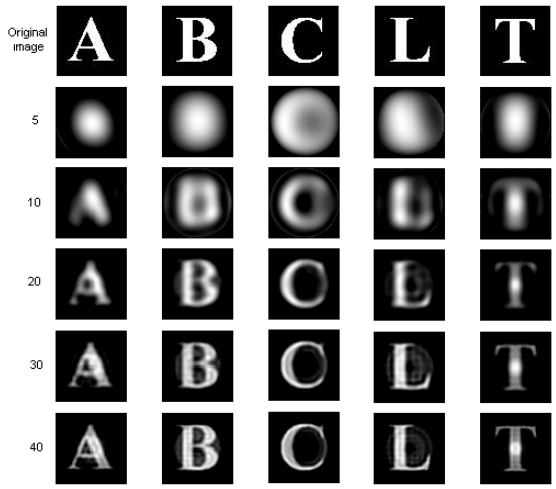


Figure 13: Zernike moment reconstruction (Image from http://www.discover.uottawa.ca/~qchen/my_presentations/master_defense.pdf)

especially for higher order moments, make them impractical for real time operations. Another disadvantage of moments is their failure to encode certain shape symmetries. Image intensities evenly distributed around the center of mass will cancel out and have a moment of zero. Note the intensity variations in the reconstructed images from Figure 13 these are a result of symmetries not encoded into the Zernike moment. The human silhouette has some symmetry as it proceeds through the walking cycle.

For a fixed order the size of the Zernike moment feature is also fixed. For our neural network application this is an advantage in that it provides fixed length feature vectors that encode image information. The issue to resolve is, what order Zernike moments do we computer? The implementation indicates an order of 30 is sufficient to characterize the different shape configurations in most images. For this work we

experiment with 20^{th}, 30^{th} and 40^{th} order Zernike moments to fully characterize the higher order moment time and accuracy in pose recognition. The Matlab implementation utilized in this work is indicated in [2].

3.4 Contour Distance and Angle

The second most used feature, behind the silhouette, is the contour which is the outline of the silhouette. The procedure for extraction of the contour features begins with the silhouettes. A beginning point is chosen. In our case the top of the head is an easily defined point. The algorithm, chain code, follows the contour around the silhouette at each step encoding a distance and direction for the next step. This procedure is continued until the algorithm arrives at the beginning point again.

The distance from starting point and the turning angle of the contour can be computed from the chain code. The distance and turning angle for a simple object is shown in Figure 14. The distance is continuous and always returns to zero. The turning angle is discontinuous.

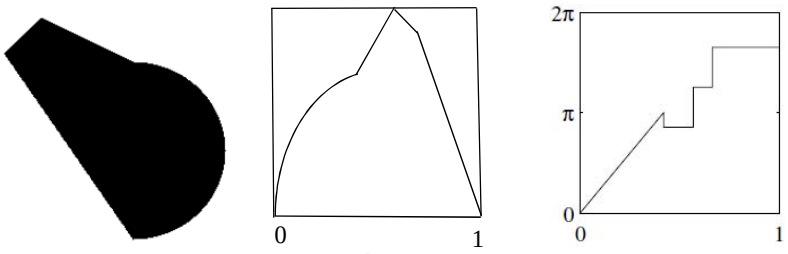
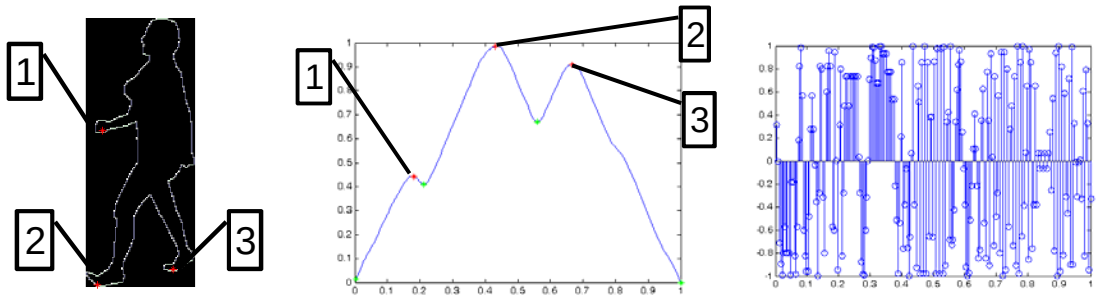


Figure 14: Distance and turning angle representation starting at bottom proceeding counter clockwise (a) Shape (b) Distance (c) Turning angle. (Image adapted from [7])

The distance and turning angle of a human silhouettes is shown in Figure 15(a),(b),(c). From this representation more advanced information can be obtained. After the distance

curve is smoothed the minimum and maximum points can be obtained as shown in Figure 15(b). The maximum points can be shown to represent appendages as indicated in Figure 15(a). The contour distance and turning angle encode the silhouette shape.



(a) (b) (c)

Figure 15: Distance and turning angle representation starting at top proceeding counter-clockwise (a) Silhouette (b) Distance (c) Turning angle.

To create fixed length feature vectors for all images the distance and turning angle are normalized. From the normalized distance and turning angle the feature vector is extracted. Even sampling over the normalized graphs provides the contour distance and angle feature vector. The Matlab chain code algorithm utilized in this work, to extract the contour, is indicated in [7].

Contour features preclude any information from the inside or outside of the silhouette. Holes inside the image silhouette may provide additional information about limb location that is lost to the contour. This algorithm also requires continuous silhouettes. An appendage that is disjointed from the main silhouette, due to bad segmentation, will not become part of the contour information. Synthetic models create ideal contours, but real life contours are rarely this clean.

3.5 Speeded Up Robust Features (SURF)

3.5.1 Overview

SURF features [9] are a recently developed interest point detector and descriptor similar in function to the SIFT [8] feature. These interest point features utilize algorithms to detect and describe local features in images. Interest point features are used primarily in the computer vision task of object recognition. SURF was developed partly as a fast approximation of the SIFT descriptor. Interest point features may utilize drastically different algorithms but generally have three main components. First they detect the location of interest points in an image. There are a wide range of interest point detectors that can detect different image features such as corners or blobs. Second the interest point features describe the local area about the interest points they detect. And third they match different images based upon the descriptors they extract. SURF and SIFT are the two main interest point features in use today.

SURF is several times faster than SIFT and claimed by its authors to be more robust against different image transformations than SIFT. Both use a blob interest point detector. As a basic image interest point detector SURF uses a Haar wavelet approximation of the determinant of Hessian Figure 16(a). SIFT uses a difference of Gaussians as an image interest point detector. An example of the interest points detected by SURF is shown in Figure 16(c). Both SURF and SIFT use a descriptor that is a spatial distribution of the gradient around the interest point. The SURF local descriptor is based on sums of approximated 2D Haar wavelets Figure 16(b)(d) in the neighborhood of the interest points. SIFT utilizes a HOG descriptor to describe the local area around the

interest points it identifies. In this respect these interest point features act very similar to the human eye. They distinguish interest points and they are sensitive to contrast.

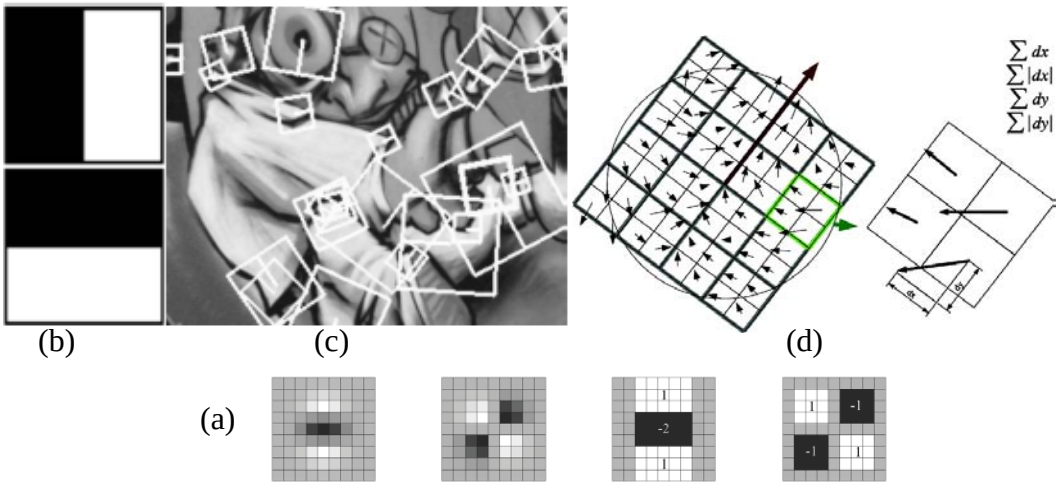


Figure 16: Details and example of SURF features (a) Box filter approximation for interest point detection (b) Harr wavelet (c) Example (d) Descriptor details (Images from [9])

SURF relies on the integral image to provide fast robust features. The integral image, also called a summed area table, is an image of the accumulated intensities from the top left of the image to the bottom right. The advantage of an integral image is that it allows for summation of the intensities in a sub region of the image with three algebraic operations as indicated in Figure 17(a). This fast calculation is utilized in both the interest point detector and the descriptor of the SURF feature.

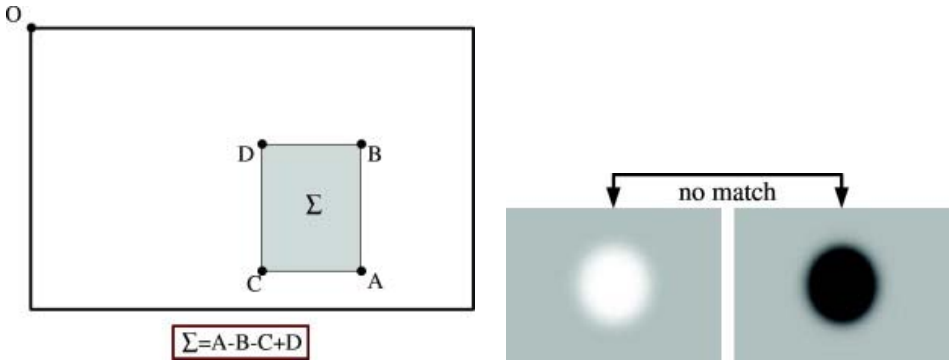


Figure 17: Details of SURF extraction (a) Integral image (b) Matching (Images from [9])

For the interest point detector, of the SURF feature, an approximation of the determinate of Hessian filter is used. This filter is applied to the entire image and is efficiently implemented with the integral image. For scale invariance the filter size is adjusted and the interest point detection algorithm is repeated. This provides a set of distinctive points that are repeatable over a wide range of image transformations.

The SURF descriptor is a collection of Harr wavelet responses around the interest point as shown in Figure 16(d). The local area is divided into a 4x4 grid. The *x* and *y* components of the Harr wavelet response and the absolute value are calculated for each of the four quadrants of the local area. This provides a descriptor with length of 64 numbers, for each interest point. This descriptor is distinctive and robust to noise, detection displacement and geometric and photometric deformation.

During calculation of the interest point location and local area descriptor three other piece of information is obtained about the interest points. The polarity of the interest point detected is indicated by the sign of the Laplacian (i.e. the trace of the Hessian matrix) for each underlying interest point. Typically, the interest points are found at blob-type structures. The sign of the Laplacian distinguishes bright blobs on dark backgrounds from the reverse situation as shown in Figure 17(b). This is used for fast matching of image descriptors. In our case this indicates whether or not the interest point is inside or outside the silhouette. Also calculated for each interest point are; the strength of the response to the interest point detector, the angle of the local area descriptor, and the scale at which the interest was detected.

For SURF features to be of use in pose recognition they must first be shown to consistently select interest points relevant to our task. In experiments upon silhouettes

SURF features consistently selected the head and appendages as interest points. A comparison of the silhouettes and the SURF features show a clear walking pattern could be discerned as shown in Figure 18. As with the silhouette the SURF features are inconsistent when faced with self occlusion.

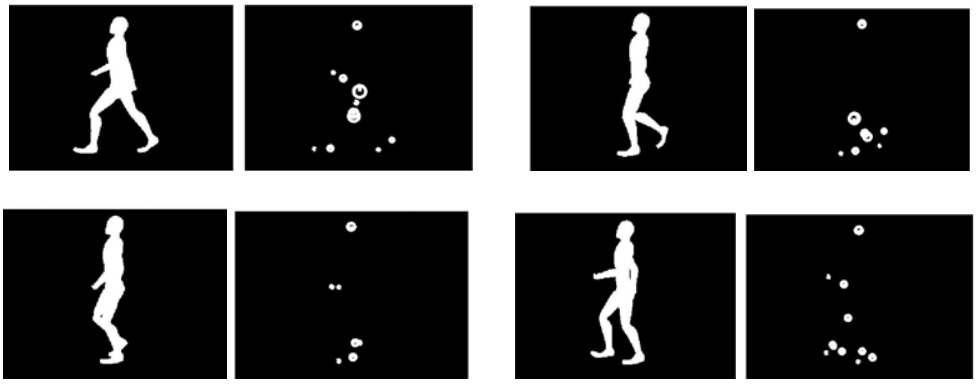


Figure 18: Silhouette and matching SURF interest point locations

The major issue in developing a feature vector using SURF features is the variability of the number of interest points detected in images. Over all the images in the walking cycle anywhere from 3 to 19 interest points are detected in any image. To create a feature vector that can be utilized by our neural networks the number of interest points in each image needs to be fixed, because the size of the descriptor for each interest point is fixed. Three separate SURF feature extraction algorithms were developed to fix the feature vector length. SURF1, SURF2, and SURF3 are developed with increasing interest point size 2, 10, and 30 interest points respectively.

3.5.1 SURF1 Features

The first fixed length SURF feature reduces each individual image regular SURF features down to two interest points. The interest points in an image can be divided into

two groups those inside the silhouette and those outside. SURF1 is a simple descriptor average of these two groups of descriptors. The interest points located inside the silhouette and outside the silhouette are separated by using the sign of the Laplacian discussed earlier. The average of all interest point descriptors inside the silhouette creates one descriptor, and the average of all the interest point descriptors outside the silhouette creates another descriptor. These two descriptors are combined to form one fixed feature vector for each image as shown in Figure 19.

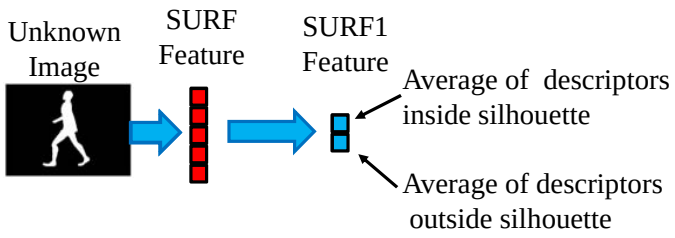


Figure 19: Procedure for extraction of SURF1 feature vectors

3.5.3 SURF2, SURF3 Features

The second SURF feature SURF2 developed for neural networks are a feature vector of 10 SURF interest point descriptors. First to create this feature vector all image SURF features, for a particular view, are combined into one feature by, their similarity and spatial proximity. A base vector is created from 10 SURF descriptors that are in the majority of the images as shown in Figure 20. These 10 SURF descriptors become the SURF2 base set for that view. To extract the SURF2 features for a particular image first the image regular SURF features are extracted. Than the image SURF interest point descriptors are matched to the SURF2 base set. The matches in the base set are replaced with the image descriptors they match Figure 21. This ensures a feature vector of 10 descriptors that contains the image information.

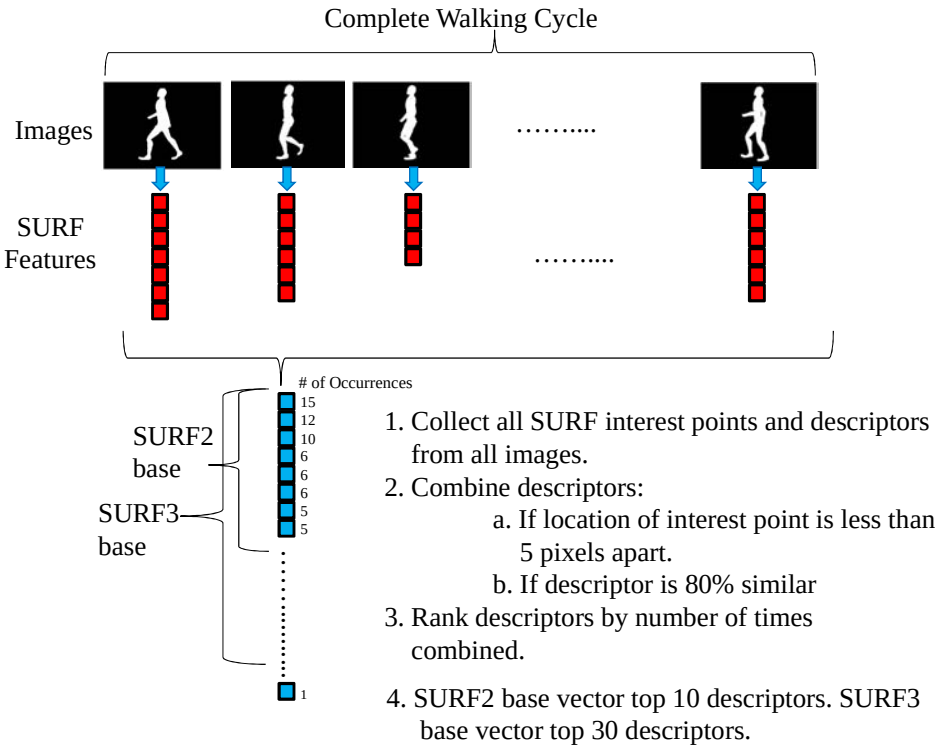


Figure 20: Procedure for creation of SURF2 and SURF3 base vectors

In this manner the image SURF features are projected onto the SURF2 base set. Ten descriptors were chosen as an average of the number of interest points in any particular image. Some images will contain more interest points than the SURF2 base set. The SURF2 feature for images with more than 10 interest points will lose information.

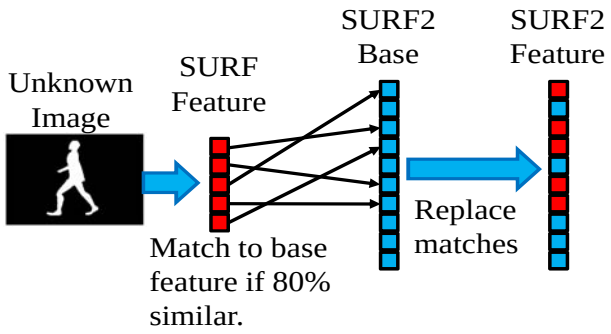


Figure 21: Procedure for extraction of SURF2 feature for unknown image.

To explore the possible value of a larger base vector the SURF3 feature vector was developed. The third SURF feature SURF3 is very similar to SURF2 but the number of base descriptors is increased to 30 as indicated in Figure 20. With thirty interest points in the base vector all SURF interest points in an image should be incorporated into the SURF3 feature, but discriminative power may be lost due to the information added to images with fewer interest points.

CHAPTER 4

NEURAL NETWORKS

4.1 Overview

Although artificial neural networks do not reach the complexity of even the simplest biological brain, they have found many applications in industry and research. Artificial neural networks like their biological counterparts consist of simulated neurons interconnected in layers. The strength of the connections between neurons and the arrangement of the neurons determine the function of the artificial neural network.

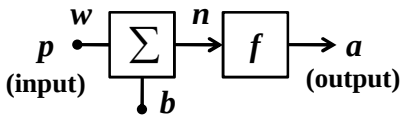


Figure 22: One artificial neuron.

A basic artificial neuron is shown in Figure 22 the equation for this neuron is.

$$a = f(wp + b), \tag{4.1}$$

The output of the neuron, *a*, is determined by the two adjustable parameters *w*, weight and *b*, bias, and the transfer function *f*. A designer of a neural network selects a transfer function and interconnection of the neurons for a specific purpose. Then the *w* and *b* parameters are adjusted through the learning process, to meet a specified input/output relationship.

The learning process is the key step in designing an accurate neural network. The learning process for neural networks proceeds in an iterative fashion as illustrated in Figure 23. First we create a set of training input/target vectors to indicate the performance of the network we desire. The training inputs are presented to the neural network with, initial guesses of the adjustable parameters x_0. The output is then compared to the targets.

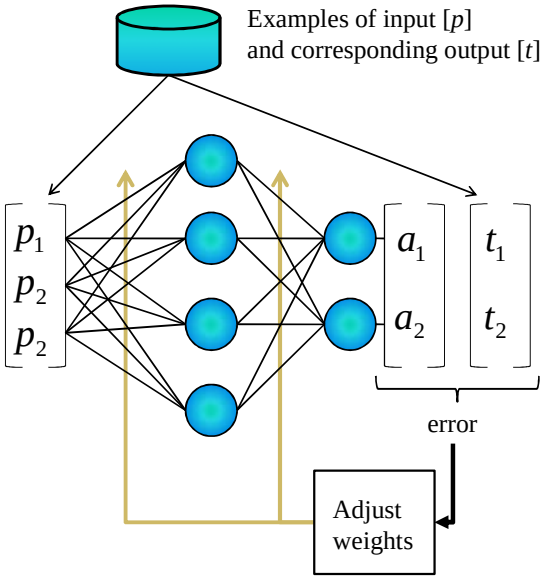


Figure 23: Training process for neural networks.

The comparison used is mean squared error between the output and the targets; this produces a performance index *F(x)* over the domain of *x* the adjustable parameters. The goal of training is to optimize, find the minimum value of *F(x)*, by adjusting *x* in successive iterations. The problem of optimizing a function has a long theoretical history stretching back to the seventeenth century. The majority of approaches to optimizing a function center around determining the first order derivative, gradient, or the second order derivative, Hessian, of *F(x)* and adjusting x_{n+1} to decrease $F(x_{n+1})$. An idealized representation of this learning process is shown in Figure 24. The performance index *F(x)* is optimized by following its gradient to the minimum.

The steepest descent algorithm utilizes the gradient to optimize *F(x)*, although the simplest technique it exhibits long training times. Newton's method utilizes the Hessian

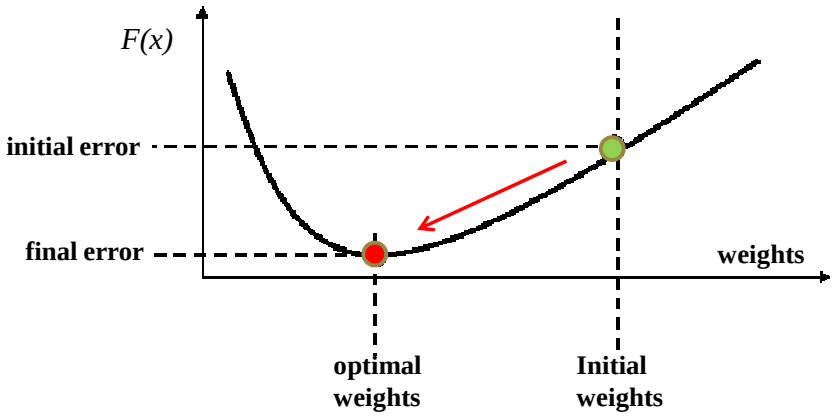


Figure 24: Ideal neural network learning.(Image adapted from http://www.nexyad.net/HTML/e-book-Tutorial-Neural-Networks.html)

and converges to an optimized solution much faster but requires calculation and storage of the Hessian which may be impractical for large networks. The conjugate gradient algorithm is a fast algorithm that does not require calculation and storage of the Hessian.

4.2 Conjugate Gradient Algorithm

The conjugate gradient algorithm for adjusting the weight space for a single layer neural network follows. First the initial search direction d_0 is chosen to be the negative of the initial gradient g_0.

$$d_0 = -g_0, \tag{4.2}$$

Next the parameters x_k are adjusted according to Equation 4.3. The learning rate α_k is chosen to minimize *F(x)* in direction d_k.

$$x_{k+1} = x_k + \alpha_k d_k, \tag{4.3}$$

Then the next search direction is determined according to Equation 4.4.

$$d_k = -g_k + \beta_k d_{k-1}, \tag{4.4}$$

Where β_k is calculated by one of the methods in Equation 4.5. This guarantees that d_k is orthogonal to Δg_k from equation 4.6 , for quadratic *F(x).*

$$\beta_k = \begin{cases} \dfrac{\Delta g_{k-1}^T g_k}{\Delta g_{k-1}^T d_{k-1}} \\ \dfrac{g_k^T g_k}{g_{k-1}^T g_{k-1}} \\ \dfrac{\Delta g_{k-1}^T g_k}{g_{k-1}^T g_{k-1}} \end{cases}, \tag{4.5}$$

$$g_k \equiv \nabla F(x)\big|_{x=x_k} \qquad \Delta g_k = g_{k+1} - g_k, \tag{4.6}$$

The above algorithm is for single layer neural networks which can only solve linearly separable problems. Three further steps need to be applied to utilize multi layer neural networks that can solve non linear problems, like pose recognition.

The neural network must be generalized, to multiple layers, through a process called back propagation. In a single layer network the calculation of the performance function derivatives is a direct function of the adjustable parameters. For multilayer networks the direct relationship between network weights and performance function are altered. Back propagation allows for derivative calculations utilizing the chain rule.

$$\frac{df(n(w))}{dw} = \frac{df(n)}{dn} \times \frac{dn(w)}{dw}, \tag{4.7}$$

Two other problems remain to fully implement a multilayer neural network. New methods to find and converge on the local minimum are applied. Second the exact

minimum may not be reached in a finite number of steps. To solve this problem the search direction is reset to the steepest descent direction after a set number of iterations.

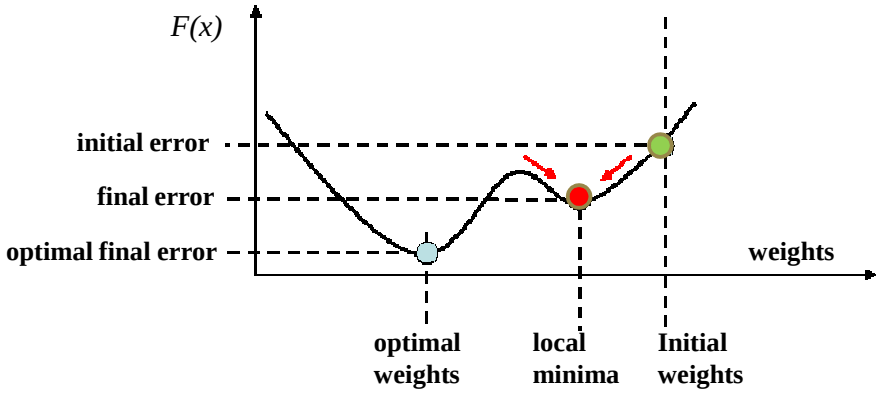


Figure 25: Neural network learning with local minima (Image adapted from http://www.nexyad.net/HTML/e-book-Tutorial-Neural-Networks.html).

The main problem encountered, with multilayer neural networks, is their susceptible to local minimum as shown in Figure 25. Many techniques have been developed to overcome this problem. In this work we employ a stochastic method. Multiple networks with different initial weights are trained. The neural networks with the smallest final mean squared error are than utilized for the purpose of testing, and all other networks discarded. The outputs, of the remaining neural networks, are then averaged for a final neural network output. Although a powerful tool in solving complex nonlinear problems close attention must be paid to avoid local minima that can reduce performance.

4.3 Matlab Implementation

Matlab is utilized throughout this work to provide a consistent platform for our feature evaluation. Our engine for pose recognition is a series of trained artificial neural networks. Utilizing the first three synthetic models, the input feature vectors and associated output target poses are generated. This training set is then used to train 100

neural networks. As the networks are trained the local minima neural networks are discarded. For each batch of five networks, the four with the largest mean squared error are discarded. This creates a total of twenty final networks for each feature and in each of the twelve views. The implementation utilizes the Matlab Neural Network Toolbox. A typical training run dialog box is shown in Figure 26.

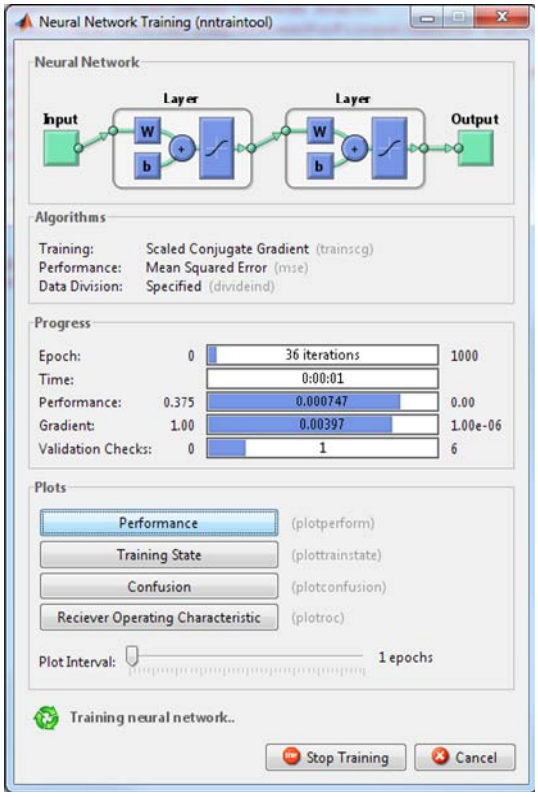


Figure 26: Matlab neural network training dialog box

For each training session the input and targets vectors are randomly divided into three sets. 60% of the vectors are used to train the network. 20% of the vectors are used to validate how well the network generalized. The remaining 20% are used as a testing set. Training continues as long as the training reduces the network's error on the validation set. The training can also end if the number of iterations exceeds the maximum

of 1000. After the network learns the training set, at the expense of generalizing more poorly, training is stopped. This technique avoids the problem of overfitting, which can otherwise foul the learning algorithm.

Finally, the last two synthetic models are used for testing. The feature vectors and pose targets provide an independent test of network generalization to data that the network has never seen. The feature vectors are applied to the twenty neural networks for each feature and view and the output, of the 20 neural networks, is averaged. View pose accuracy is calculated in a binary fashion. An image is correctly classified if it contains only true positives and true negatives pose classifications. Images with any false positive or false negative pose classifications are deemed incorrect.

CHAPTER 5

RESULTS

5.1 Experimental Setup

A set of experiments were conducted to test the pose classification accuracy of different features while utilizing neural network. A diverse set of algorithms extracted pose and feature information from video of synthetic models walking. The algorithm was implemented with Matlab running on a PC with Pentium IV 3.2 GHz CPU and 1GB RAM. The experiments were based upon training a set of neural networks with data from the first three synthetic models shown in Figure 3 and testing with data from the last two synthetic models shown in Figure 3. Additionally the HumanEva dataset [20] was used for testing. The results of these experiments are presented.

Experiment 1 is a comparison of pose classification accuracy for all features in the full walking cycle. The results of experiment 1 are displayed in Table 1 for each of the 12 views, and in Table 2 for each of the 10 poses. This is a test of the discriminative capabilities of each of the features in the task of human pose classification utilizing neural networks, synthetic models and modern features.

Experiment 2 is a comparison of pose classification accuracy for all features in the half walking cycle. The results of experiment 2 are displayed in Table 3 for each of

the 12 views. The half walking cycle provides higher accuracy because it avoids the left right ambiguity associated with the full walking cycle silhouettes.

Experiment 3 is a comparison of pose classification accuracy for the three developed SURF features SURF1, SURF2, and SURF3 in the half walking cycle. The results of experiment 3 are displayed in Table 4 for each of the 12 views. This is a test of the discriminative capabilities of each of the developed features in each of the 12 views.

Experiment 4 utilizes the HumanEva dataset [20] in a comparison of pose classification accuracy for real life testing data for all features. The results of experiment 4 are displayed in Table 5 for each of the 12 views. This is a test of the discriminative capabilities of the synthetically trained neural networks upon real life training data. Real life Human Eva dataset images are classified into one of 5 poses in the half walking cycle.

Experiment 5 compares all features with their advantages and disadvantages. The results of the overall comparison are presented in Table 6. The advantages and disadvantages of different features are discussed

5.2 Data Collection

The data for these experiments began as video of the synthetic models completing one full walking cycle. Multiple videos of the each model, one for each of the 12 view directions in Figure 4, were generated. Each video was split into 10 evenly spaced segments that represent the 10 poses in the complete walking cycle. The video is separated into individual images by extracting 10 frames for each of the 10 pose segments. With 5 models, 12 views, 10 poses, and 10 frames per pose we extract a total

of 6,000 video frame images from our original videos data. These 6,000 synthetic model walking images are the raw image dataset for this research.

The raw image dataset is further processed to create the final silhouette image dataset that features are extracted from. Each image in the raw image dataset is first segmented into foreground and background areas by assigning a 1 to foreground pixels and a 0 to background pixels. The original raw image data is resized to 110x110 pixels from the original raw image size of 320x240. 110x110 pixels are chosen as a standard silhouettes image size for feature extraction because it provides a compact representation of the human shape.

The HumanEva dataset [20] contains video of humans walking a circle associated with detailed 3D pose information. A similar process to the one above was used to obtain 110x110 pixel silhouettes from the Human Eva dataset as shown in Figure 27. This process created a set of 801 silhouette images with pose information.

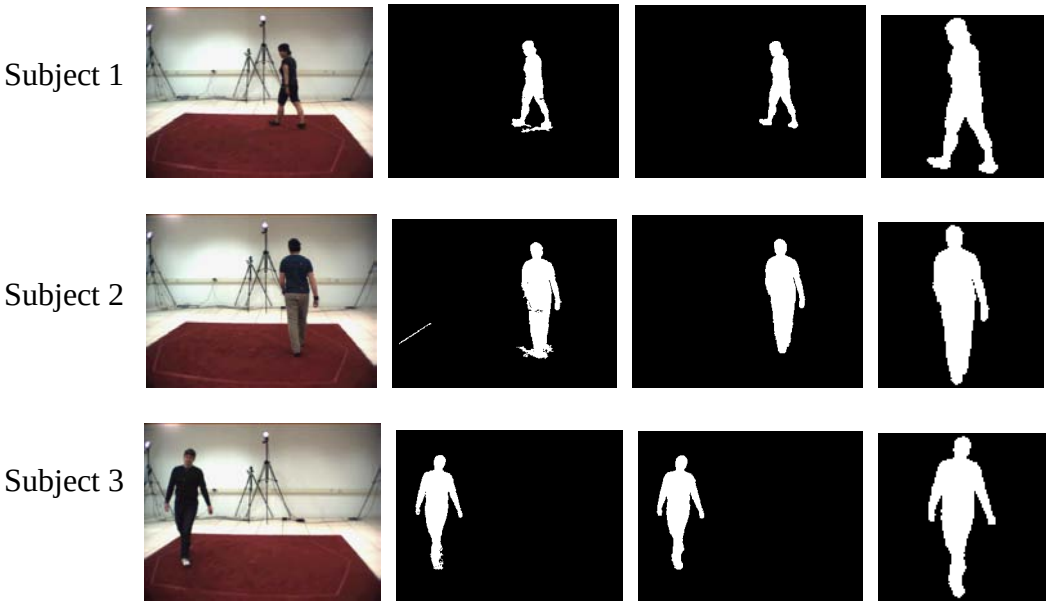


Figure 27: HumanEva data (a) Video image (b) Background segmentation (c) Manually cleaned segmentation (d) Final 110x110 binary silhouette.

5.3 Experiment 1 Full Walking Cycle Feature Accuracy

Table 1 indicates the results of experiment 1 the accuracy of our neural networks in the full walking cycle pose classification task are presented. The dense features show the greatest performance. This result shows the value of the HOG feature to extract the most information from the images. The view also contributes a great deal to pose classification accuracy with partial side views 3, 5, 9, and 11 containing more discriminative information and higher classification accuracy. The walking toward view 1 and walking away view 7 produced the worst accuracy results. The higher 40^{th} order Zernike moment has lower accuracy than the 30^{th} order moment. The 30^{th} order moment is 3% more accurate than the 20^{th} and 40^{th} order moments. The spares features contour distance, turning angle, and SURF had the lowest accuracy. PCA reduced the accuracy of the silhouette and HOG feature by 15~20%.

Table 1: Accuracy full walking cycle for synthetic models by view direction.

Pose Classification Accuracy by View and Feature Full Walking Cycle												
View	HOG	Zernike Moment (30)	Silhouette	Silhouette (Distance Transform)	Zernike Moment(20)	Zernike Moment (40)	PCA (Silhouette)	PCA (HOG)	Contour Distance and Angle	SURF1	Contour Distance	Avg.
1	70%	66%	60%	56%	58%	66%	49%	50%	46%	70%	49%	58%
2	87%	87%	82%	79%	81%	88%	76%	82%	60%	74%	59%	77%
3	90%	91%	91%	89%	90%	86%	90%	76%	69%	53%	61%	80%
4	87%	85%	88%	88%	85%	79%	80%	61%	59%	57%	57%	75%
5	91%	85%	88%	91%	79%	84%	85%	77%	79%	64%	78%	82%
6	86%	84%	84%	90%	82%	78%	75%	78%	68%	56%	68%	77%
7	78%	67%	60%	61%	62%	67%	43%	51%	24%	55%	23%	54%
8	88%	84%	87%	86%	88%	82%	77%	73%	60%	52%	54%	75%
9	90%	93%	92%	90%	91%	90%	89%	71%	68%	63%	63%	81%
10	89%	89%	90%	87%	85%	81%	82%	68%	73%	59%	67%	79%
11	94%	87%	92%	92%	88%	87%	86%	77%	74%	57%	75%	82%
12	92%	87%	88%	83%	88%	86%	81%	80%	69%	63%	68%	80%
Avg.	87%	84%	83%	83%	81%	81%	76%	70%	62%	60%	60%	
Key	Best ----------->Worst											

Table 2 from experiment 1 details the accuracy of the pose classification for each of the 10 poses in the full walking cycle. Overall this table indicates that view has more of an impact than pose upon classification accuracy. The poses at the furthest ends of the walking cycle pose 1 and 10 are the most discriminative and the ones in the center are the least discriminative. These three poses are represented by a silhouette with the legs furthest apart in the contact pose. Poses 1 and 10 contain the most information and are the most discriminative. The converse is true of the middle of the walking cycle. The accuracy of the neural network classifier is reduced to its lowest level for pose 6.

Table 2: Accuracy full walking cycle for synthetic models by pose.

Pose Classification Accuracy by Pose and Feature Full Walking Cycle												
Pose	HOG	Zernike Moment (30)	Silhouette	Silhouette (Distance Transform)	Zernike Moment(20)	Zernike Moment (40)	PCA (Silhouette)	PCA (HOG)	Contour Distance and Angle	SURF1	Contour Distance	Avg.
1	95%	92%	92%	93%	87%	89%	88%	72%	74%	52%	75%	82%
2	84%	86%	78%	80%	82%	78%	73%	70%	62%	55%	58%	73%
3	88%	85%	82%	86%	82%	83%	75%	71%	72%	67%	65%	78%
4	90%	86%	88%	86%	83%	88%	80%	76%	63%	71%	61%	79%
5	77%	89%	82%	76%	85%	85%	70%	63%	65%	55%	66%	74%
6	82%	74%	78%	81%	70%	73%	66%	53%	34%	42%	36%	63%
7	81%	80%	79%	80%	80%	78%	75%	68%	41%	47%	36%	68%
8	92%	82%	82%	78%	78%	78%	74%	75%	61%	61%	59%	74%
9	86%	75%	78%	78%	77%	71%	75%	73%	75%	68%	73%	75%
10	91%	88%	93%	89%	88%	88%	83%	82%	73%	82%	70%	84%
Avg	87%	84%	83%	83%	81%	81%	76%	70%	62%	60%	60%	
Key	Best ----------->Worst											

5.4 Experiment 2 Half Walking Cycle Feature Accuracy

The results of experiment 2 are contained in Table 3. Table 3 indicates accuracy of our neural networks in the half walking cycle pose classification task for each view. The dense features show the greatest performance. The view contributes a great deal to accuracy results with side views containing far more discriminative feature vectors for the neural network. For the half walking cycle the 30^{th} and 40^{th} order Zernike moments

performed the best. The HOG, and binary silhouette also perform well. Contour distance and turning angle again performed the worst. The SURF1, SURF2, and SURF3 designed feature performance for the half walking cycle are detailed in section 5.5. The dimensional reduction PCA only reduced the accuracy 2~5% for the half walking cycle. The accuracy is again particularly bad for view 1 and view 7 with their limited pose information.

Table 3: Accuracy half walking cycle for synthetic models by view direction.

Pose Classification Accuracy by View and Feature Half Walking Cycle											
View	Zernike Moment (30)	Zernike Moment (40)	HOG (Histogram of Oriented Gradient)	Silhouette	Zernike Moment (20)	PCA (Silhouette)	Silhouette (Distance Transform)	PCA (HOG)	Contour Distance and Angle	Contour Distance	Avg.
1	89%	89%	69%	79%	85%	66%	66%	65%	61%	69%	74%
2	90%	93%	92%	90%	90%	86%	91%	85%	77%	77%	87%
3	91%	92%	92%	92%	93%	94%	92%	88%	88%	88%	91%
4	93%	95%	94%	97%	94%	95%	99%	83%	87%	86%	92%
5	95%	95%	99%	94%	90%	93%	95%	93%	89%	90%	93%
6	95%	89%	91%	95%	92%	88%	96%	92%	79%	78%	90%
7	81%	87%	76%	76%	72%	70%	57%	71%	21%	14%	63%
8	88%	83%	93%	88%	91%	91%	84%	87%	78%	77%	86%
9	95%	93%	90%	92%	89%	94%	94%	82%	80%	79%	89%
10	93%	92%	96%	96%	93%	95%	92%	91%	83%	82%	91%
11	94%	92%	95%	95%	92%	95%	93%	90%	89%	90%	93%
12	90%	93%	88%	91%	89%	89%	88%	89%	82%	84%	88%
Avg.	91%	91%	90%	90%	89%	88%	87%	85%	76%	76%	
Key	Best----------Worst										

5.5 Experiment 3 SURF1 SURF2 SURF3 Feature Accuracy

Table 4 indicates the accuracy results for the three developed SURF features SURF1, SURF2, and SURF3. Although the average accuracy for all three SURF features is about equal there is useful information in the table. The highest accuracy is indicated for each view. In general classification accuracy for views that SURF1 and SURF2 performed poorly showed improved performance with the SURF3 feature. This indicates that a SURF feature that combines the best attributes SURF1, SURF2, and SURF3 could

be more discriminative than individual SURF features. This experiment seems to indicate that a smaller fixed length SURF feature like SURF1 and SURF2 create more accurate results for views with fewer image SURF interest points like views 1 and 7. Larger SURF features like SURF3 create more accurate pose classification for views with more interest points, but lower accuracy for views with fewer interest points.

Table 4: Accuracy half walking cycle for SURF1, SURF2, and SURF3 features.

Pose Classification Accuracy Half Cycle SURF Features by View				
View	SURF1	SURF2	SURF3	Avg.
1	75%	72%	55%	67%
2	51%	50%	63%	55%
3	59%	60%	81%	67%
4	37%	37%	64%	46%
5	66%	61%	72%	66%
6	67%	69%	62%	66%
7	71%	63%	38%	57%
8	61%	56%	61%	59%
9	66%	71%	48%	62%
10	60%	61%	71%	64%
11	55%	57%	47%	53%
12	58%	57%	57%	57%
Avg.	61%	60%	60%	

5.6 Experiment 4 HumanEva Dataset Accuracy

In an attempt to test the generalization our results pose classification accuracy of real life images from the HumanEva dataset are tested. This experiment indicates the effect training only with synthetic models on the task of pose classification of the half walking cycle. Table 5 indicates the results of the pose classification accuracy measured. The general poor results indicate the pitfalls of testing on data that a model has not been trained upon. The HumanEva dataset also contains continuous view direction change not the discrete views of the synthetic models as shown in Figure 27. Real life data represent a serious problem for idealized models built upon synthetic data. The silhouette

performed the best. The 30th order Zernike moment and HOG performed the worst indicating the value of the binary silhouette feature. This also indicates that if a close correspondence between the training and testing data does not exist, HOG features will perform badly. The silhouette is more robust to changes in testing data.

Table 5: Accuracy half walking cycle for HumanEva dataset by view.

Pose Classification Accuracy by View and Feature HumanEva Dataset									
View	Silhouette	Zernike Moment (40)	Zernike Moment (20)	Silhouette (Distance Transform)	Contour Distance and Angle	Contour Distance	HOG (Histogram of Oriented Gradient)	Zernike Moment (30)	Avg.
1	0%	0%	0%	0%	6%	8%	4%	0%	2%
2	14%	36%	0%	0%	7%	7%	0%	0%	7%
3	30%	16%	10%	13%	26%	25%	8%	13%	16%
4	7%	3%	24%	3%	21%	17%	17%	10%	14%
5	33%	14%	20%	24%	22%	18%	24%	20%	20%
6	42%	39%	39%	42%	39%	39%	3%	28%	33%
7	0%	6%	0%	0%	0%	0%	0%	6%	2%
8	27%	37%	40%	20%	20%	17%	27%	33%	28%
9	3%	0%	0%	0%	10%	19%	3%	0%	5%
10	35%	17%	40%	35%	6%	8%	17%	31%	22%
11	19%	22%	16%	25%	28%	28%	16%	13%	21%
12	17%	10%	0%	26%	10%	7%	36%	2%	13%
Avg.	19%	17%	16%	16%	16%	16%	13%	13%	
Key	Best----------Worst								

5.7 Experiment 5 Overall Feature Characteristics

Different features have different strengths and weaknesses. Size, speed of calculation, and accuracy in determining pose are all important attributes of features. Table 1 lists the size, speed, accuracy, and general characteristics of the different features utilized. The average accuracy is calculated from Experiment 1 and 2. The largest three numbers are indicated in each column. The table shows that the dense features outperform the sparse features at the cost of increased memory use. The HOG feature is half the size of the silhouette and the distance transform with equal or improved accuracy performance. The table also shows that for a 10 percent decrease in accuracy PCA can be

employed. Of the dense features Zernike moments maintained the high accuracy of the dense features with drastically reduced memory requirements.

Of the sparse features the contour distance and angle are the most discriminative, with drastically reduced accuracy. Contour distance and turning angle, are the simplest and most discriminative sparse feature that can be calculated fast. The disadvantage of the contour is the requirement of a clean well segmented input image which may not be possible for real world examples. The remaining sparse features SIFT and SURF indicates the motivation for this research they have small size can handle noisy disjoint silhouettes, and at least in the case of SURF are fast. The major problem of SURF features is the variability of the feature size.

Table 6: Overall comparison of features

Comparison of Features					
Feature	size(bytes)	Speed (seconds for 50 images)	handle disjoint	Dense/Sparse	Average accuracy
Silhouette	96,800	1.84	yes	dense	87%
Silhouette (PCA)	~2,000	1.84	yes	dense	82%
Distance Transform	96,800	2.23	yes	dense	85%
HOG	48,672	2.23	yes	dense	89%
HOG (PCA)	~2,000	2.23	yes	dense	78%
Zernike Moments(20)	1,936	120.43	yes	dense	85%
Zernike Moments(30)	4,096	364.92	yes	dense	88%
Zernike Moments(40)	7,056	849.53	yes	dense	86%
Contour distance	2,000	4.63	no	sparse	68%
Contour distance/angle	4,000	5.09	no	sparse	69%
SIFT	2,000-8,000	253.45	yes	sparse	n/a
SURF	2,000-8,000	5.13	yes	sparse	61%

CHAPTER 6

CONCLUSION AND FUTURE RESEARCH

6.1 Conclusions

Five experiments were performed to test the discriminative value of a wide variety of image feature vectors in the task of human pose recognition. These experiments allow objective evaluation of different image feature vectors. Modern image features such as HOG and SURF can be compared to binary image silhouettes, silhouette distance transform, contour features, and image Zernike moments. The pose classification accuracy of these features and modified versions of these features can be tested to indicate their relative discriminative power for pose recognition. Image feature vectors that are of fixed length can be tested for their pose classification accuracy in this manner. Three fixed length SURF related features are developed for use with our neural network pose classifier. Additionally linear dimensional reduction, PCA, was performed upon the binary silhouette and the HOG features, and the resulting vectors treated as separate feature vectors to determine their discriminative power.

The HOG features showed the overall best performance for the task of full walking cycle pose recognition as indicated by Experiment 1. Experiment 1 also showed the effect of PCA upon feature pose classification accuracy. A 15% to 20% drop in pose

classification accuracy is indicated for features that undergo PCA dimensional reduction. This indicates that the nonlinear discriminative information in the feature data is lost in the linear process of PCA dimensional reduction. The binary silhouette showed overall good performance for pose recognition. The silhouette distance transform feature showed no classification accuracy improvement over the binary silhouette feature. The silhouette distance transform didn't provide any additional information about pose than the binary silhouette, for our neural network classifier. Zernike moments were shown to be a compact and accurate feature as indicated in Table 1,2,3 and 5, but computationally expensive as indicated in Table 6. The contour distance and turning angle features were shown to be compact, easy to calculate, but inaccurate for pose classification. The contour features have limited discriminative ability because they are limited to describing the edge of the human silhouette. Important discriminative information inside and outside the silhouette are lost to the contour features. All features except SURF1 and SURF2 performed poorly on views of the human silhouette as they walk toward, view 1, and walk away, view 7. These views are the most difficult to accurately classify pose.

No feature is optimal, different features may be advantageous in different applications. The sparse features tested, contour and SURF performed less accurately than the dense features in pose classification accuracy due to lack of spatial information, but were the most compact as indicated in Table 6. The variable number of SURF interest points for silhouettes in different poses and different views make correlation of SURF features difficult. SURF1 and SURF2 features were designed with low fixed feature vector length. These features performed well on views with a low number of interest points, view 1 and 7, as indicated in Table 4. SURF3 features are designed with a high

fixed feature vector length. This feature performed well on views with a high number of interest points, view 3 and 10, as indicated in Table 4. None of the designed SURF features had high overall pose recognition accuracy. Table 4 also indicates the strength of the SURF features, in that they extract the minimum discriminative information from an image. This opens the possibility of designing features that are compact and discriminative for all views with SURF features.

Testing with real life data, HumanEva dataset, is indicated in Experiment 4, using the neural networks trained upon synthetic models didn't produce promising results. The neural network discriminative model is highly sensitive to the training data, and does not generalize well to real life data once trained upon synthetic data. The experiment indicated that the binary silhouette is the most robust of the features tested on real life data. Experiment 4 also indicates the pitfalls of training on limited data of a specific kind. The trained neural networks are discriminating upon features that are specific to synthetic models not to human poses in real life images.

6.2 Future Work

This system could be used with additional features or modifications of existing features. As new features appear they can be trained and tested in the same procedure described above to gage the discriminative capabilities of the new feature. Furthermore many of the presented features can be optimized by altering feature extraction parameters. For example the HOG feature cell or block dimensions can be altered, or the order of the Zernike moments can be optimized to gain the highest accuracy.

Training and testing with real life data could be implemented. The testing of real life data with the neural networks trained upon synthetic models is inadequate for true pose recognition. The synthetic models could also be altered to be more like real life data. Synthetic models could be driven with different human gaits, noise could be added to the silhouette images, or a wider array of synthetic human body types could be utilized for training.

Additionally actual images could be utilized for feature extraction to avoid some of the ambiguities associated with silhouettes. The modern features utilized SURF and HOG are intended for actual images not silhouettes. In the future more image features will be developed for object recognition in real images and these features should be utilized to their fullest in the task of pose recognition. These features could recover some of the information lost in the process of going from image to silhouette.

Additional SURF feature algorithms could be devised and tested. The increase in the number of base descriptors in SURF3 increased the accuracy for views that tested poorly for SURF2 and SURF1. As indicated in Table 4 the SURF2 and SURF3 features could be combined into a feature vector with increased overall pose classification accuracy.

REFERENCES

[1] M. K. Hu, "Visual pattern recognition by moment invariants," IRE Trans. Inf. Theory IT-8, 179-187 (1962).

[2] M.R. Teague, “Image analysis via the general theory of moments”. J. Opt. Soc. Am. **70** (1980), pp. 920–930. (Matlab implementation http://www.lans.ece.utexas.edu/~lans/lans /lanstoolbox.zip)

[3] N. Dalal and B, Triggs, “Histograms of Oriented Gradients for Human Detection” In CVPR 2005 (Matlab implementation http://www.shamoxia.com/html/y2010/1690.html)

[4] H.Fujiyoshi and A. Lipton, "Real-time Human Motion Analysis by image Skeletonization", IEEE Workshop on Applications of Computer Vision, p15-21, Princeton, 1998.

[5] M.T. Hagan, H.B. Demuth and M.H. Beale “Neural Network Design”, PWS Publishing, Boston, MA (1996).

[6] Rogez, G. Rihan, J. Ramalingam S. Orrite C., Torr P.H.S,. “Randomized trees for human pose detection”. In Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, June2008.

[7] Nicholas R. Howe “Silhouette Lookup for Automatic Pose Tracking”, IEEE Workshop on Articulated and Nonrigid Motion. 2004 (Matlab implementation http://maven.smith.edu/~nhowe/research/code)

[8] Lowe, David G. "Object recognition from local scale-invariant features". Proceedings of the International Conference on Computer Vision. **2**. pp. 1150–1157.. 1999.790410

[9] Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346--359, 2008 [8] Hagan, Demuth, Beale ,"Neural Network Design”,.(Matlab implementation http://www.vision.ee.ethz.ch/~surf/) (Images from http://cvrr.ucsd.edu/ece285/papers/bay06_surf.pdf)

[10] Dariu M. Gavrila. “The visual analysis of human movement: A survey”. Computer Vision and Image Understanding (CVIU), 73(1):82–92, January 1999.

[11] Jake K. Aggarwal and Qin Cai. “Human motion analysis: a review”. Computer Vision and Image Understanding (CVIU), 73(3):428–440, March 1999.

[12] Liang Wang, Weiming Hu, and Tieniu Tan. “Recent developments in human motion analysis”. Pattern Recognition, 36(3):585–601, March 2003.

[13] David A. Forsyth, Okan Arikan, Leslie Ikemoto, James O'Brien, and Deva Ramanan." Computational Studies of Human Motion Part 1: Tracking and Motion Synthesis". Foundations and Trends in Computer Graphics and Vision, 1(2):77–254, July 2006.

[14] Elden Yu and J.K. Aggarwal "Human Action Recognition with Extremities as Semantic Posture Representation", International Workshop on Semantic Learning Applications in Multimedia in association with CVPR, 2009

[15] Li-Qun Xu and David C. Hogg. "Neural networks in human motion tracking - an experimental study". Image and Vision Computing, 15(8):607–615, August 1997.

[16] Ankur Agarwal and Bill Triggs. "Recovering 3D human pose from monocular images".IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 28(1):44–58, January 2006.

[17] Ronald Poppe. "Vision Based Human Motion analysis: An Overview". Computer Vision and Image Understanding (CVIU), 73(3):4–18, Nov 2007.

[18] David A. Forsyth, Jean Ponce, "Computer Vision: A Modern Approach". Prentice Hall, 2002

[19] Rafael C. Gonzalez, Richard E. Woods, “Digital Image Processing”, Prentice Hall, 2002 (p34-45)

[20] Leonid Sigal and Michael J. Black. “HumanEva: Synchronized Video and Motion Capture Dataset for Evaluation of Articulated Human Motion”. Technical Report CS-06-08, Brown University, Department of Computer Science, Providence, RI, September 2006.

VITA

Michael Kruis

Candidate for the Degree of

Master of Science

Thesis: HUMAN POSE RECOGNITION USING NEURAL NETWORKS, SYNTHETIC MODELS, AND MODERN FEATURES

Major Field: Electrical Engineering

Biographical:

Education:
Completed the requirements for the Master of Science in Electrical Engineering Department of Oklahoma State University, Stillwater, Oklahoma in July, 2010.

Experience:

Spring 2010: Teaching Assistant for Signal Analysis
Spring 2010: Research Assistant VCIPL

Professional Memberships:

Student member of IEEE

Name: Michael Kruis Date of Degree: July, 2010

Institution: Oklahoma State University Location: Stillwater, Oklahoma

Title of Study: HUMAN POSE RECOGNITION USING NEURAL NETWORKS, SYNTHETIC MODELS, AND MODERN FEATURES

Pages in Study: 64 Candidate for the Degree of Master of Science

Major Field: Electrical Engineering

Scope and Method of Study: Computer Vision

Findings and Conclusions:

Our goal in this research is to compare modern image feature vectors with traditional image feature vectors in the task of human pose recognition. Recently newer image feature vectors such as Histograms of Oriented Gradient (HOG's) and Speeded Up Robust Features (SURF) have been successfully utilized for object recognition in images. The value of these newer feature vectors compared to traditional feature vectors for pose recognition has not been fully addressed. Our study uses synthetic human animation models, neural networks, and a variety of image feature vectors for pose recognition.

In this approach feature vectors and pose information are extracted from five 3D human gait animations created from five human models. We define 10 poses in a full walking cycle and 12 views around the human model. Ten images are extracted for each pose, view and model, resulting in total 6000 images (3600 for training and 2400 for testing). Features are divided into dense and sparse representations. The former one includes binary silhouette, distance transform of silhouette, HOG's, and Zernike moments, and the latter one embraces contour distance, contour angle, and SURF. Moreover, three SURF related fixed length feature vectors are developed. A set of neural networks are then trained to match the feature/pose relationship specified by the extracted data.

The HOG feature proved to be the best overall feature for pose recognition with the highest pose recognition accuracy. High accuracy for SURF features could not be achieved with fixed length SURF features. High accuracy for individual views and specific SURF feature lengths was shown. The silhouette feature is shown to be robust and effective in general. Zernike moments are compact and highly accurate at pose recognition, but required a long computational time. Contour features were low in accuracy but easy to extract and compact. The silhouette distance transform did not perform significantly better than the silhouette. We also discuss the advantages and disadvantages of individual feature in this work.

ADVISER'S APPROVAL: Dr. Guoliang Fan